

**PEMBANGUNAN PERANGKAT LUNAK
ERROR MONITORING PADA WEB UNTUK MENUNJANG
PERAWATAN PERANGKAT LUNAK
(Studi Kasus PT. Integra Solusi Teknotama)**

TUGAS AKHIR

Disusun sebagai salah satu syarat untuk kelulusan Program Strata 1,
di Program Studi Teknik Informatika, Universitas Pasundan Bandung

oleh :

Galih Riyadi
NRP : 13.304.0295



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN BANDUNG
MARET 2019**

LEMBAR PENGESAHAN LAPORAN TUGAS AKHIR

Telah diujikan dan dipertahankan dalam Sidang Sarjana Program Studi Teknik Informatika Universitas Pasundan Bandung, pada hari dan tanggal sidang sesuai berita acara sidang, tugas akhir dari :

Nama : Galih Riyadi
Nrp : 13.304.0295

Dengan judul :

**“PEMBANGUNAN PERANGKAT LUNAK *ERROR MONITORING* PADA WEB UNTUK
MENUNJANG PERAWATAN PERANGKAT LUNAK”**



Pembimbing Utama,

Pembimbing Pendamping,

(Dr.Ir.Leony Lidya, MT.)

(Wanda Gusdya, S.T., MT)

DAFTAR ISI

ABSTRAK.....	i
ABSTRACT.....	ii
KATA PENGANTAR	iii
DAFTAR ISI.....	iv
DAFTAR ISTILAH	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR	x
DAFTAR SIMBOL.....	xii
BAB 1 PENDAHULUAN.....	1-1
1.1 Latar Belakang	1-1
1.2 Identifikasi Masalah	1-2
1.3 Tujuan Tugas Akhir.....	1-2
1.4 Lingkup Tugas Akhir	1-2
1.5 Metodologi Tugas Akhir	1-3
1.6 Sistematika Penulisan	1-4
BAB 2 LANDASAN TEORI	2-1
2.1 Rekayasa Perangkat Lunak (<i>Software Engineering</i>).....	2-1
2.1.1 Proses Perangkat Lunak	2-1
2.1.2 Model Proses Generik	2-3
2.2 Software Maintenance	2-5
2.2.1 <i>Maintenance Process</i>	2-6
2.3 <i>Software Error</i>	2-7
2.4 <i>PHP Errors</i>	2-12
2.4.1 <i>PHP 5 Error Levels</i>	2-12
2.4.2 Codeigniter Error Handling	2-14
2.5 <i>Web Information Exchange Diagram (WIED)</i>	2-15
2.5.1 Pendekatan Pada Aspek Informasi	2-15
2.5.2 Spesifikasi Notasi <i>WIED</i>	2-17
2.6 <i>Application Programming Interface (API)</i>	2-19
2.7 Penelitian Terdahulu.....	2-22
BAB 3 SKEMA PENELITIAN.....	3-1
3.1 Kerangka Tugas Akhir	3-1
3.2 Analisis Masalah dan Solusi TA.....	3-3
3.3 Kerangka Pemikiran Teoritis	3-4
3.4 Tempat dan Objek Penelitian.....	3-6

3.4.1 Tempat Penelitian.....	3-6
3.4.2 Objek Penelitian	3-6
BAB 4 ANALISIS DAN PERANCANGAN	4-1
4.1 Analisis Sistem yang Berjalan	4-1
4.2 Gagasan Sistem Usulan	4-1
4.3 Alur Aktifitas.....	4-2
4.4 Analisis <i>User</i> Yang Terlibat	4-3
4.5 Analisis Kebutuhan Fungsional	4-3
4.6 Analisis Kebutuhan <i>Non-Fungsional</i>	4-4
4.7 Diagram <i>Use Case Web Error Monitoring</i>	4-4
4.8 Skenario <i>Use Case</i>	4-6
4.8.1 Skenario <i>Use case</i> Melakukan <i>Reporting Bug</i>	4-6
4.8.2 Skenario <i>Use Case</i> Menampilkan <i>Error</i> dan <i>Bug</i>	4-7
4.8.3 Skenario <i>Use Case</i> Melakukan Validasi terhadap <i>Error</i> Serta <i>Bug</i>	4-7
4.8.4 Skenario <i>Use Case</i> Alokasi Sumber Daya.....	4-8
4.8.5 Skenario <i>Use Case</i> Mengarsipkan Riwayat Perbaikan	4-8
4.8.6 Skenario <i>Use Case</i> Melakukan Validasi Perbaikan <i>Error</i>	4-9
4.9 Diagram <i>Sequence</i>	4-10
4.9.1 Diagram <i>Sequence</i> Melakukan <i>Reporting Bug</i>	4-10
4.9.2 Diagram <i>Sequence</i> Menampilkan <i>Error</i> dan <i>Bug</i>	4-11
4.9.3 Diagram <i>Sequence</i> Melakukan Validasi Terhadap <i>Error</i> Serta <i>Bug</i>	4-12
4.9.4 Diagram <i>Sequence</i> Alokasi Sumber Daya	4-13
4.9.5 Diagram <i>Sequence</i> Mengarsipkan Riwayat Perbaikan	4-14
4.9.6 <i>Sequence</i> Melakukan Validasi Perbaikan <i>Error</i>	4-15
4.10 <i>Mockup</i> Perangkat Lunak	4-16
4.10.1 <i>Mockup</i> Antarmuka Pemimpin Proyek	4-16
4.10.2 <i>Mockup</i> Antarmuka <i>Programmer</i>	4-22
4.10.3 <i>Mockup</i> Klien	4-28
4.10.4 <i>Mockup</i> Tambahan.....	4-28
4.11 <i>Web Information Exchange Diagrams (WIED)</i>	4-31
4.12 Perancangan Basis Data.....	4-33
4.13 Perancangan Diagram Kelas.....	4-33
BAB 5 IMPLEMENTASI DAN PENGUJIAN	5-1
5.1 Implementasi Perangkat Lunak	5-1
5.1.1 Kebutuhan Perangkat Lunak	5-1
5.1.2 Kebutuhan Perangkat Keras (<i>Hardware</i>).....	5-2
5.1.3 Penggunaan <i>Framework</i>	5-2

5.2 Implementasi Basis Data	5-4
5.3 Implementasi Web API	5-5
5.3.1 Struktur Kelas	5-5
5.3.2 Endpoints	5-6
5.4 Implementasi Antarmuka	5-23
5.4.1 Antarmuka Pemimpin Proyek	5-23
5.4.2 Antarmuka Programmer	5-28
5.5 Implementasi Error Handling	5-32
5.6 Pengujian	5-34
5.6.1 Rencana Pengujian	5-34
5.6.2 Metode Pengujian	5-34
5.6.3 Fungsionalitas Perangkat Lunak	5-34
5.6.4 Skenario Pengujian	5-36
5.6.5 Pengujian Perangkat Lunak	5-37
5.6.6 Kesimpulan User Testing	5-37
BAB 6 KESIMPULAN DAN SARAN	6-1
6.1 Kesimpulan	6-1
6.2 Saran	6-2
6.3 Rekomendasi	6-2
DAFTAR PUSTAKA	i
LAMPIRAN	L

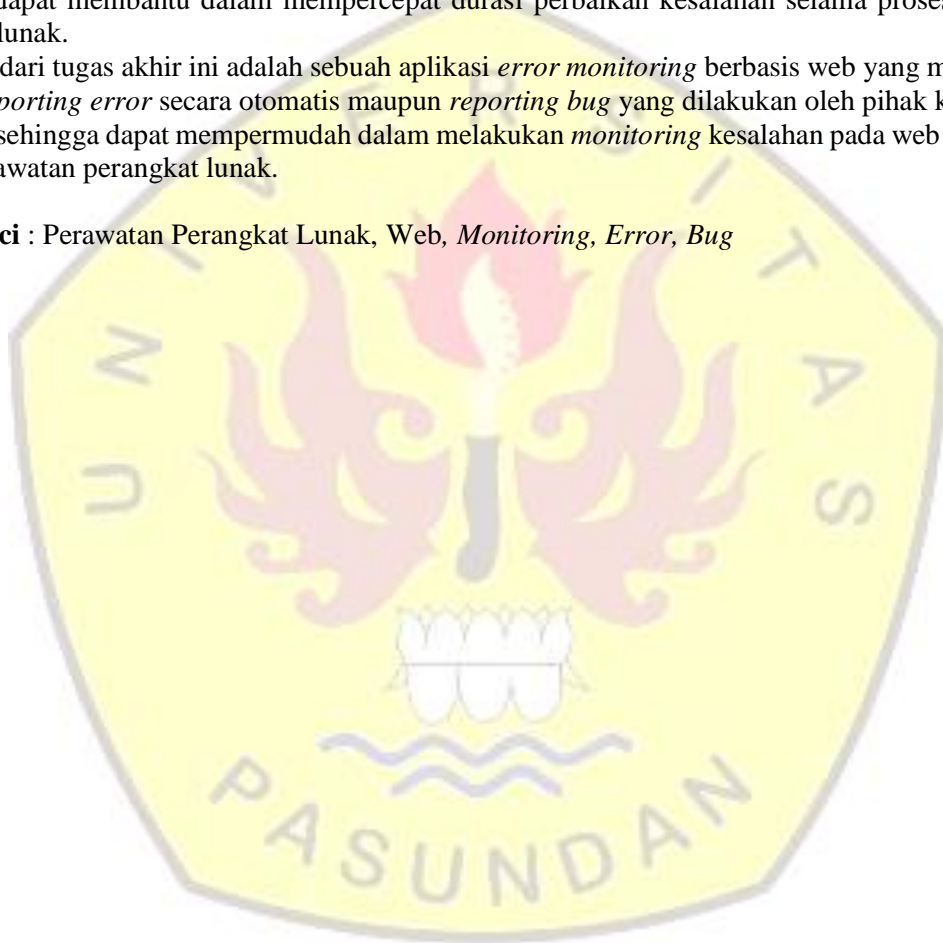
ABSTRAK

Perawatan perangkat lunak adalah proses memodifikasi sistem perangkat lunak atau komponen setelah *delivery*, untuk memperbaiki kesalahan, memperbaiki kinerja, atau untuk menyesuaikan diri dengan lingkungan yang berubah. Untuk dapat menemukan dan memperbaiki kesalahan yang terjadi pada web klien maka sebelumnya perlu dilakukan kegiatan *monitoring* secara berkala kepada setiap web klien tersebut, mencatat rincian kesalahan yang ditemukan, dan selanjutnya melakukan perbaikan oleh pihak *developer*.

Untuk dapat menyelesaikan masalah yang ada, maka dilakukan proses pembangunan aplikasi menggunakan metode *Software Development Life Cycle* yang berfungsi melakukan *monitoring* terhadap *error* maupun *bug* pada web klien untuk mempermudah dalam proses perbaikan perangkat lunak yang dilakukan oleh *developer*. Durasi dalam proses perbaikan akan sangat berpengaruh terhadap waktu ketersediaan layanan pada web-web klien yang bersangkutan, untuk itu diharapkan aplikasi yang dibangun dapat membantu dalam mempercepat durasi perbaikan kesalahan selama proses perawatan perangkat lunak.

Hasil dari tugas akhir ini adalah sebuah aplikasi *error monitoring* berbasis web yang menyediakan layanan *reporting error* secara otomatis maupun *reporting bug* yang dilakukan oleh pihak klien kepada *developer* sehingga dapat mempermudah dalam melakukan *monitoring* kesalahan pada web klien dalam proses perawatan perangkat lunak.

Kata Kunci : Perawatan Perangkat Lunak, Web, *Monitoring*, *Error*, *Bug*



ABSTRACT

Software maintenance is the process of modifying a software system or component after delivery, to correct errors, improve performance, or adjust to a changing environment. To be able to find and correct errors that occur on the web client, it is necessary to conduct periodic monitoring activities for each web client, record the details of errors found, and then make improvements by the developer.

To be able to solve the existing problem, the application development process is carried out using the Software Development Life Cycle method which functions to monitor errors and bugs on the web client to simplify the process of software improvement carried out by the developer. The duration of the improvement process will greatly affect the time of availability of services on the web of the clients concerned, so that applications that are built can be expected to help in accelerating the duration of repairing errors during the software maintenance process.

The result of this final project is a web-based error monitoring application that provides reporting error services automatically and reporting bugs carried out by the client to the developer so that it can simplify monitoring errors on the web client in the software maintenance process.

Keywords: Software Maintenance, Web, Monitoring, Error, Bug



BAB 1

PENDAHULUAN

Pada bab 1 akan menjelaskan mengenai latar belakang identifikasi masalah, tujuan Tugas Akhir, lingkup Tugas Akhir, metodologi Tugas Akhir, dan sistematika penulisan Tugas Akhir.

1.1 Latar Belakang

Institute of Electrical and Electronics Engineers (IEEE) mendefinisikan perawatan perangkat lunak adalah proses memodifikasi sistem perangkat lunak atau komponen setelah *delivery*, untuk memperbaiki kesalahan, memperbaiki kinerja, atau untuk menyesuaikan diri dengan lingkungan yang berubah [THO01]. Untuk melakukan perawatan perangkat lunak, maka perlu dilakukannya peninjauan(*monitoring*) berkala terhadap perangkat lunak yang berjalan, terutama aspek *error* dan *bug* pada perangkat lunak tersebut. Sehingga, dapat diambil tindakan tertentu dari hasil *monitoring* tersebut, serta dapat dengan segera dilakukannya penanganan terhadap permasalahan-permasalahan yang ditemukan.

Misalnya, pada perusahaan yang bergerak dibidang *Software Developer*, membuat lebih dari satu produk *web* terhadap klien yang berbeda. Setelah aplikasi *web* di-*deliver* kepada klien, pihak *developer* menginginkan untuk dapat melakukan *monitoring* secara berkala atau setidaknya mendapatkan *report* dengan cepat terhadap masalah yang terjadi terhadap perangkat lunak tersebut. Maka untuk memecahkan masalah tersebut, pihak *developer* harus melakukan *monitoring* setiap saat untuk mengetahui jika sewaktu-waktu terjadi *error* ataupun *bug*, sehingga hal ini akan menjadi semakin sulit jika hal tersebut dilakukan terhadap lebih dari satu aplikasi *web*.

Dampak perkembangan teknologi saat ini dapat menjadikan banyak hal dimudahkan, tentu saja hal ini dapat membantu banyak pihak dalam menjalankan bisnisnya ataupun aktivitas-aktivitas lainnya. Seperti halnya dalam proses *report error* ataupun *bug* pada aplikasi *web* dan juga aktivitas komunikasi dengan klien dapat sangat terbantu dengan dukungan teknologi. Sebelumnya, semua proses *komplain* atau *report* klien terhadap masalah pada aplikasi *web* berupa *error* ataupun *bug* dilakukan kepada *developer* melalui *e-mail* ataupun melakukan pertemuan secara langsung. Akibatnya, pihak *developer* akan kesulitan jika komunikasi maupun *report* terhadap *error* atau *bug* pada aplikasi *web* dilakukan kepada lebih dari satu klien. Sehingga, dari pihak *developer* memerlukan waktu yang relatif lebih lama dalam proses distribusi informasi dari pemimpin proyek kepada *programmer* yang menangani masing-masing aplikasi *web* tersebut. Untuk itu, yang perlu dilakukan adalah membuat teknologi yang dapat membantu dalam menangani masalah-masalah tersebut, yaitu perangkat lunak yang dapat mengirimkan *report bug*, *error* ataupun *log* jika diperlukan antara aplikasi *web* klien kepada pihak *developer*.

Sehingga, pihak *developer* mempunyai akses terpusat mengenai masalah *error*, *bug* ataupun aktivitas lain pada aplikasi *web* yang telah dibuat, serta memungkinkan pihak administrator atau pengelola dari aplikasi *web* yang dimiliki klien mengirimkan *report bug* atau *error* yang ditemukan pada aplikasi *web*, kemudian *report* dari klien tersebut dikelompokkan berdasarkan masing-masing aplikasi *web* yang bersangkutan, Sehingga dapat lebih memudahkan *developer* untuk memproses semua *report*

yang ada. Dari sisi lain, *monitoring* yang dilakukan, dapat membantu mengetahui masalah atau hal-hal lainnya yang diperlukan oleh pihak *developer* untuk memudahkan dalam proses *maintenance*.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah dipaparkan sebelumnya, maka permasalahan yang dimunculkan pada tugas akhir ini adalah :

1. Bagaimana membuat sebuah perangkat lunak yang dapat me-*monitoring web-web* klien untuk dapat mempercepat proses perbaikan pada kegiatan perawatan perangkat lunak(*maintenance*).
2. Bagaimana cara mengelompokkan *report bug* dan *error* dari semua klien agar dapat memudahkan dalam pembagian tugas dalam proses perbaikan.
3. Bagaimana membuat sebuah perangkat lunak yang dapat menghubungkan pihak *developer* dengan banyak aplikasi *web* yang dimiliki oleh klien.

1.3 Tujuan Tugas Akhir

Tujuan tugas akhir ini adalah :

1. Dihasilkannya perangkat lunak yang dapat melakukan *monitoring* dan *reporting error* serta *bug* yang terjadi pada aplikasi *web* yang dimiliki klien, sehingga dapat mempercepat proses perbaikan pada perawatan perangkat lunak(*maintenance*).
2. Dihasilkannya perangkat lunak yang mampu melakukan alokasi terhadap penanganan *error* serta *bug* yang ditemukan.

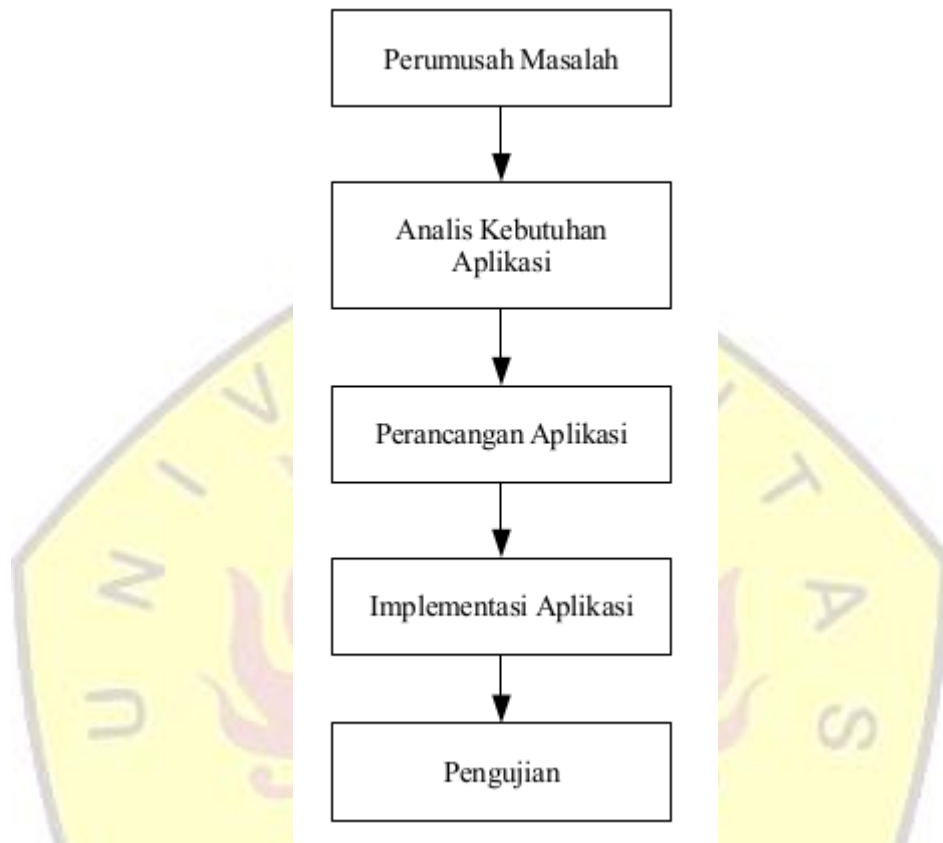
1.4 Lingkup Tugas Akhir

Adapun lingkup dari Tugas Akhir ini adalah:

1. Penelitian pengumpulan data dan studi kasus dilakukan di PT. Integra Solusi Teknotama.
2. Fokus penelitian adalah fase perbaikan *error* serta *bug* pada proses perawatan perangkat lunak.
3. Perangkat lunak yang dibangun adalah *web error monitoring* dan *web API* sebagai penghubung semua aplikasi *web* klien.

1.5 Metodologi Tugas Akhir

Metodologi merupakan kerangka dasar dari tahapan penyelesaian tugas akhir. Metodologi penulisan pada tugas akhir ini mencakup semua kegiatan yang dilaksanakan untuk memecahkan masalah atau melakukan proses analisa terhadap permasalahan tugas akhir. Berikut ini merupakan metodologi tugas akhir, dapat dilihat pada gambar 1.1 dibawah ini.



Gambar 1. 1. Metodologi Tugas Akhir

Berikut merupakan rincian dari metodologi tugas akhir ini, diantaranya adalah :

1. Perumusan Masalah
Pada tahap ini dilakukan mengidentifikasi masalah. Mengkaji aktivitas sebenarnya serta memutuskan dengan pasti masalah yang akan disederhanakan dalam bentuk aplikasi.
2. Analisis Kebutuhan Aplikasi
Pada tahap ini dilakukan untuk menentukan *requirement* yang dibutuhkan terhadap aplikasi.
3. Perancangan Aplikasi
Pada tahap ini dilakukan perancangan aplikasi berdasarkan dengan analisis kebutuhan.
4. Implementasi Aplikasi
Pada tahap ini dilakukan implementasi dari hasil perancangan yang sebelumnya telah dibuat.
5. Pengujian
Pada tahap ini dilakukan pengujian sistem serta *feedback* yang diberikan oleh pihak yang bersangkutan terhadap aplikasi yang telah dibuat.

1.6 Sistematika Penulisan

Dalam penulisan Tugas Akhir ini, penulis menyusun pembahasan menjadi beberapa bab sebagai berikut :

BAB 1 PENDAHULUAN

Merupakan bab pendahuluan yang berisi garis besar permasalahan yang akan dibahas dan diselesaikan sesuai dengan tujuan yang telah dirumuskan seperti latar belakang penelitian, identifikasi masalah, tujuan tugas akhir, lingkup tugas akhir, batasan tugas akhir, metodologi tugas akhir dan sistematika penulisan tugas akhir.

BAB 2 LANDASAN TEORI

Pada bab ini berisi penjelasan mengenai teori-teori dasar yang berkaitan dengan studi kasus yang dilakukan dalam penelitian Tugas Akhir yaitu mengenai perangkat lunak *web error monitoring*, dan *bug reporting* serta *Web API* sebagai penghubung semua aplikasi web yang ada.

BAB 3 SKEMA PENELITIAN

Pada bab ini berisi mengenai kerangka penyelesaian Tugas Akhir, rencana dan alur analisis yang akan dilakukan, analisis peta dan relevansi penggunaan konsep atau teori, dan tempat penelitian.

BAB 4 ANALISIS DAN PERANCANGAN

Pada bab ini berisi uraian analisis dan perancangan dari perangkat lunak *web monitoring* yang akan dibuat sesuai dengan metode yang digunakan.

BAB 5 IMPLEMENTASI DAN PENGUJIAN

Pada bab ini berisi mengenai implementasi dan pengujian terhadap aplikasi *web error monitoring* beserta *Web API* berdasarkan pada hasil analisis dan perancangan sebelumnya.

BAB 6 KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan yang diperoleh dari pengerjaan Tugas Akhir serta saran yang diberikan dalam pembangunan aplikasi *web error monitoring* terhadap aplikasi *web* klien.

DAFTAR PUSTAKA

- [APP17] Appleboy., “CodeIgniter-Log-Library”, tersedia : Februari 2019, <http://github.com/appleboy/CodeIgniter-Log-Library/commits/master>, Feb 2017.
- [BRI18] British Columbia Institute of Technology., “Error Handling”, tersedia : November 2018, www.codeigniter.com/user_guide/general/errors.html, Jun 2018.
- [CON00] Conallen, J., "Building Web Applications with UML", Addison-Wesley, Boston, 2000.
- [DAN04] Daniel, Galin., "Software Quality Assurance-From theory to implementation", Addison-Wesley, Harlow, 2004
- [IEE93] Institute of Electrical and Electronics Engineers., "IEEE Standards Collection: Software Engineering", IEEE Standard 610.12-1990, IEEE, 1993.s
- [JAC12] Jacobson, Daniel., Brail, Greg., Woods, Dan., 2012. "APIs: A Strategy Guide", First Edition, o'reilly, Sebastopol, 2012.
- [KEL95] Kelleher, Kevin, Casey G., Lois D., et al, “Cause and Effect Diagram : Plain & Simple”, Joiner Associates Inc, USA, 1995.
- [MAH16] Mahendra, Rizqa G., " Simulasi Deteksi Bit Error Menggunakan Metode Hamming Code Berbasis Web", 2016.
- [MAU18] Maulana, Rizqon S., "Pembangunan Web Service Untuk Mendukung Aplikasi Reminder Pengobatan Tuberculosis", 2018.
- [NAS12] Nasution, Iftar R., "Pembangunan Aplikasi untuk Monitoring Kinerja Web Server Apache", 2012.
- [PHP19] PHP Group., “Errors”, tersedia : Feb 2019, <http://php.net/manual/en/language.errors.basics.php>, 2019.
- [PRE09] Pressman, Roger S., Lowe, David., " Web Engineering: A Practitioner's Approach", First Edition, McGraw-Hill, New York, 2009.
- [PRE10] Pressman, Roger S., "Software Engineering: A Practitioner's Approach ", Edisi 7, McGraw-Hill, New York, 2010.
- [PRE15] Pressman, Roger S., Maxim, Bruce R., " Software Engineering: A Practitioner's Approach ", Edisi 8, McGraw-Hill, New York, 2015
- [PUT15] Putra, Sendy T., " Aplikasi Helpdesk Error Report ATM BCA Berbasis Web Pada Kantor Cabang BCA ", 2015.
- [TAM17] Tama, Andika P.Putra., "Pembuatan Web Service untuk Integrasi Aplikasi Website kecamatan Danau Teluk Dengan Website pemerintahan Kota Jambi", 2017.
- [THO01] Thomas M.P., "Software Maintenance", chapter 6, Technical Software Services (TECHSOFT). Florida, 2001
- [TON04] Tongrunrojana, R., and D. Lowe, “WIED: A Web Modelling Language for Modelling Architectural-Level Information Flows”, Texas Digital Library, vol. 5, no. 2, Oktober 2018.
- [TUT18] TutorialRepublic., “PHP 5 Error Levels”, tersedia : Feb 2019, <http://www.tutorialrepublic.com/php-reference/php-error-levels.php>, 2018.